Computer Engineering and Intelligent Systems
ISSN 2222-1719 (Paper) ISSN 2222-2863 (Online)
Vol 2, No.3

Data Hiding in Binary Images Using Orthogonal Embedding - A High Capacity Approach

Bharti Gawali[1],
Dr.Babasaheb Ambedkar Marathwada University,
Aurangabad, India
E-mail: bharti_rokade@yahoo.co.in

Rupali Kasar[2],
Dr.Babasaheb Ambedkar Marathwada University,
Aurangabad, India
E-mail: rupalikasar1@gmail.com

Abstract

The growth of high speed computer networks and the Internet, in particular, has increased the ease of Information Communication. In comparison with Analog media, Digital media offers several distinct advantages such as high quality, easy editing, high fidelity copying, compression etc. But this type advancement in the field of data communication in other sense has hiked the fear of getting the data snooped at the time of sending it from the sender to the receiver. Information Security is becoming an inseparable part of Data Communication. In order to address this Information Security, Digital Watermarking plays an important role. Watermarking Techniques are used to hide a small amount of data in such a way that no one apart from the sender and intended recipient even realizes there is a hidden message. This paper proposed a high capacity data hiding approach for binary images in morphological transform domain for authentication purpose so that the image will look unchanged to human visual systems.

Keywords: *Data hiding, digital watermarking, authentication, security, authentication, binary image*.

I. Introduction

Digital watermarking is the process of embedding information into a digital signal. The signal may be audio, pictures or video. If the signal is copied, then the information is also carried in the copy [1].A significant number of data hiding techniques have been reported in recent years in order to create robust digital watermarks. Among all the existing techniques for digital color and gray scale images, not all of them can be directly applied to binary text images. Digital watermarking techniques have been proposed for ownership protection, copy control, fingerprinting, annotation and authentication of digital media [2]. Recently authentication of digital documents has found wide applications in handwritten signatures, digital books, business documents, personal documents, maps, engineering drawings, and so on [3]–[4]. Most data-hiding techniques for binary images have focused on spatial domains, for example, choosing data-hiding locations by employing pairs of contour edge patterns [5], edge pixels and defining visual quality-preserving rules [6], [7]. However, the capacities of the existing algorithms are not large enough, especially for small images.

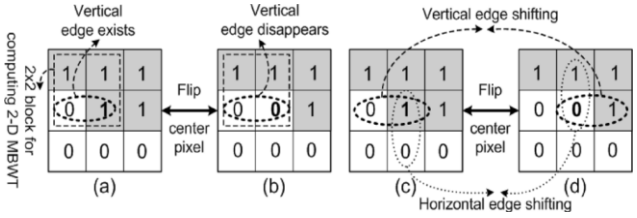


Figure.1.Illustration of changes in coefficient due to flipping pixels [(a)and(b)]and the "edge shifting "phenomenon [(c) and (d)].

A watermarking scheme is usually consist of three distinct steps,

- Embedding,
- Attack and
- Detection.

Authentication process ensures that the given set of data comes from a legitimate sender and content integrity is preserved [8]. Watermarking can be further classified into hard authentication and soft authentication [9]. Hard authentication is the validation of content that does not allow any modifications. That means a single bit change in the test image will trigger an alarm that indicates the content is unauthentic. "Hard" image authentication is highly sensitive and dependent on the exact value of image pixels, whereas "Soft" image authentication is sensitive just to content modification and serious image quality tampering.

On the other hand, editing an image becomes easier with the powerful image editing tools and digital cameras. Authentication to detect tampering and forgery is thus of primary concern. This work focuses on hard authenticator watermark-based authentication. The problem of data hiding for binary images in morphological transform domain is concentrated. The morphological binary wave transform [10] can be used to track the transitions in binary images by utilizing the detail coefficients as a location map to determine the data hiding locations. Embedding data using real-valued coefficient requires more memory space. The Morphological binary wavelet transform [10] can be used to track the transitions in binary images by using the detail coefficients. This coefficient is used as a location map to determine data hiding locations but the problem lies with the fact that once a pixel is flipped ,the horizontal, vertical and diagonal detail coefficients also changes. The major advantages of the proposed scheme lie in its larger capacity compared with previous schemes, better visual quality (e.g., compared with [5]–[9]) and lower computational complexity. In addition, unlike [4], our present scheme does not suffer the capacity decrease and computational load increase in order to incorporate the cryptographic signature for authentication.

The paper is divided into seven sections. Section I, deals with the introduction of digital watermarking. Section II, provides a brief review on existing 1-D morphological binary wavelet transform**,** our proposed extension to 2-D morphological binary wavelet transform is presented in Section III, Section IV describes the Orthogonal Embedding method, Experimental results and data hiding effects are presented In Section V and VI ,Section VII concludes the paper.

II. 1-D Morphological Binary Wavelet Transform

An interlaced transform is used to identify the embeddable locations due to the fact that some transition information is lost during the computation of a single transform and there is a need to keep track of transitions between two and three pixels for binary images data hiding. Here an image based on 2x2 pixel blocks is processed and two different processing cases are combined that the flippability conditions of one are not affected by flipping the candidates of another for data embedding are called "Orthogonal Embedding". This addresses the problem of the capacity decrease due to the un-embeddability of the block boundaries. As a result, significant gains in capacity can be achieved, which also improves the efficiency of utilizing the flippable pixels. "Exclusive OR (XOR)" operation addresses the quantization error occurred in a DCT-based approach. The major advantages of the proposed scheme lie in its larger capacity (e.g., compared with [5], [11], [12]), better visual quality (e.g., compared with [5]–[7]) and lower computational complexity (e.g., compared with [5], [4]). In addition, unlike [7], our present scheme does not suffer the capacity decrease and computational load increase in order to incorporate the cryptographic signature for authentication.

This section, reviews with signal analysis and synthesis and 1-D signal decomposition similar to [14].

Based on this, decomposition scheme to 2-D signal is extended in this paper and subsequently proposed an interlaced transform for the data-hiding application.

Consider a family of signal space V_l and detail space W_l at level l. The 1-D wavelet decomposition scheme comprises of one signal analysis operator ψ_l^+ and one detail analysis operator ϖ_l^+. In addition, it also consists of one signal synthesis operator ψ_l^- and one detail synthesis operator ϖ_l^-.Here "+" indicates that the corresponding operator maps a signal to a higher level,

towards the direction of reducing the information. Whereas "_"indicates that the operator maps a signal to a lower level, towards the direction of restoring the information. The signal analysis operator $\psi_l^+ : Vl \longrightarrow Vl+1$ maps a signal, $S_l \in V_l$ from level l to l+1 for the scaled signal $Sl+1 = \psi l + (Sl), Sl+1 \in Vl+1$ The detail analysis operator $\varpi_l^+ : V_l \longrightarrow W_{l+1}$ maps a signal from level l to l+1 for the detail signal $d_{l+1} = \varpi_l^+ (S_l)$, $d_{l+1} \in w_{l+1}$ On the other hand, the signal synthesis operator $\psi_l^- : V_{l+1} \longrightarrow V_l$ maps a signal from level l+1 to back to level l to obtain an approximation of S_l , to $S_l = \psi_l(S_{l+1})$ The detail synthesis operator $\varpi_l^-(d_{l+1})$maps a detail signal back to level l so as to obtain the detail signal $e_l = \varpi_l^-(d_{l+1})$ The signal at level l is reconstructed by

$$S_l = \psi_l^-(S_{l+1}) + \varpi_l^-(d_{l+1}) \qquad (1)$$

Perfect reconstruction of the original signal is possible if the operators satisfy

$$\psi_l^- \psi_l^+ = \varpi_l^- \varpi_l^+ = I \qquad (2)$$
$$\psi_l^- \varpi_l^+ = \varpi_l^- \psi_l^+ = \underline{0} \qquad (3)$$
$$\psi_l^- \psi_l^+ + \varpi_l^- \varpi_l^+ = I \qquad (4)$$

Where "I" and "0" represent the identity and zero operators, respectively. Note that that $\{\psi_l^-, \psi_l^+, \varpi_l^-, \varpi_l^+\}$ is a set of biorthogonal filter operators if the conditions in (2)–(4) are satisfied. For the morphological analysis and synthesis scheme the bi-orthogonality is formulated in the operator terms. Let " $\oplus$ " denote the "Exclusive OR (XOR)" operation. The signals at level l+1 by applying the analysis operators for a 1-D signal are given by

$$s(i) = \psi^+(s)(i) = s(2i+1) \qquad (5)$$
$$d(i) = \varpi^+(s)(i) = s(2i) \oplus s(2i+1) \qquad (6)$$

s(i) & d(i) are the coarse and detail signals obtained at level l+1,respectively. i denotes the index of the signal at level l+1 and i = 0,1,2,…..,N-1 for a 1-D signal of size N . The detail signal d(i) contains 1's at each transition from 0 to 1 or vice versa in signal. The synthesis operators are given by

$$\psi^-(s)\, s(2i+1) = \psi^-(s)(2i) = s(i) \qquad (7)$$
$$\varpi^-(s)\, s(2i+1) = 0 \text{ and } \varpi^-(s)\,(2i) = d(i) \qquad (8)$$

The signal at level l is reconstructed by

$$s(2i) = \psi^-(s)(2i) \oplus \varpi^-(s)\,(2i) = s(i) \oplus d(i) \qquad (9)$$
$$s(2i+1) = \psi^-(s)(2i+1) \oplus \varpi^-(s)(2i+1) = s(i) \qquad (10)$$

III. Interlaced Morphological Binary Wavelet Transform

The 1-D wavelet decomposition scheme [14] is extended to a 2-D signal by defining a non-separable 2-D transform. Let i and j represent the indices of the signal at level l +1, where i=0,1,2,…M-1 and j = 0,1,2,…N-1 for a 2-D coarse signal of size M x N .Designation of the samples in a 2 x 2 block is shown in Fig. 2.where s (2i,2j) denotes the signal ("0" or "1") located at row and column at level .To define a 2-D transform, one sample in a 2 x2 block is sub-sampled as the coarse signal.

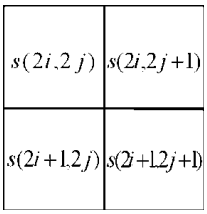


Figure 2. Designation of the samples in a 2 x 2 block

The horizontal, vertical and diagonal detail signals are derived from the differencebetween the sub-sampled sample and its vertical, horizontal and diagonal (including diagonal and anti-diagonal) neighbors. The resultant transformed signal remains binary while the coarse and detail signals will each be 1/4 the size of the original signal. Let the operators for the coarse signal, horizontal, vertical, and diagonal detail signals be, ψ^{ee}, $\varpi_h{}^{ee}$, ϖ_v^{ee} and ϖ_d^{ee}, respectively, where the superscript " **ee** " denotes "even-even". The obtained transform is named the even-even transform since it is operated on a 2 x 2 block starting from the even-even coordinates. The coarse signal, vertical, horizontal and diagonal detail signals at level $l+1$ are obtained by applying the analysis operators to obtain.

$$S^{ee}(i,j) = \psi^{ee+}(S)\,(i,j) = S(2i+1,2j+1) \quad (11)$$

$$v^{ee}(i,j) = \varpi_h{}^{ee+}(S)\,(i,j) = S(2i+1,2j) \oplus S(2i+1,2j+1) \quad (12)$$

$$h^{ee}(i,j) = \varpi^{ee+}(S)\,(i,j) = S(2i,2j+1) \oplus S(2i+1,2j+1) \quad (13)$$

$$d^{ee}(i,j) = \varpi_d{}^{ee+}(S)\,(i,j) = S(2i,2j) \oplus S(2i+1,2j) \oplus S(2i,2j+1) \oplus S(2i+1,2j+1) \quad (14)$$

Where $S^{ee}(i,j) \in V_{l+1}$ and $\{v^{ee}(i,j), h^{ee}(i,j), d^{ee}(i,j)\} \in W_{l+1}$, the synthesis operators of a 2-D wavelet transform are given by

$$\begin{aligned}\psi^{ee}(s)(2i,2j) &= \psi^{ee}(s)(2i,2j+1)\\ &= \psi^{ee}(s)(2i+1,2j)\\ &= \psi^{ee}(s)(2i+1,2j+1)\\ &= S^{ee}(i,j)\end{aligned} \quad (15)$$

Finally, the signal at level $l+1$ can be reconstructed by

$$S(2i,2j) = \psi^{ee}(s)(2i,2j) \oplus \varpi_d{}^{ee}(S)\,(2i,2j)$$

$$= S^{ee}(i,j) \oplus v^{ee}(i,j) \oplus h^{ee}(i,j) \oplus d^{ee}(i,j) \quad (16)$$

$$S(2i,2j+1) = \psi^{ee}(s)(2i,2j+1) \oplus \varpi_h{}^{ee}(S)\,(2i,2j+1)$$

$$= S^{ee}(i,j) \oplus h^{ee}(i,j) \quad (17)$$

$$S(2i+1,2j) = \psi^{ee}(s)(2i+1,2j) \oplus \varpi_v{}^{ee}(S)\,(2i+1,2j)$$

$$= S^{ee}(i,j) \oplus v^{ee}(i,j) \quad (18)$$

$$S(2i+1,2j+1) = \psi^{ee}(s)(2i+1,2j+1) \oplus \varpi^{ee}(S)(2i+1,2j+1)$$
$$= S^{ee}(i,j) \qquad (19)$$

Since the coarse signals obtained from (11) are at the odd-odd locations, the transitions from odd-odd to other coordinates in the 2 x 2 block are assessed from the detail signals obtained by (12)–(14). To obtain the transition from odd-even, even-odd and even-even coordinates complementary wavelet transforms operating on the 2 x 2 blocks is designed which start from the even-odd, odd-even and odd-odd coordinates of the signal. Based on the absolute coordinates in the top left position shown in Figure.3. Each 2 x 2 block in a 2-D image is classified as an even-even block (EEB),or an even -odd block (EOB), or an odd-even block (OEB), or an odd-odd block (OOB), which is given by

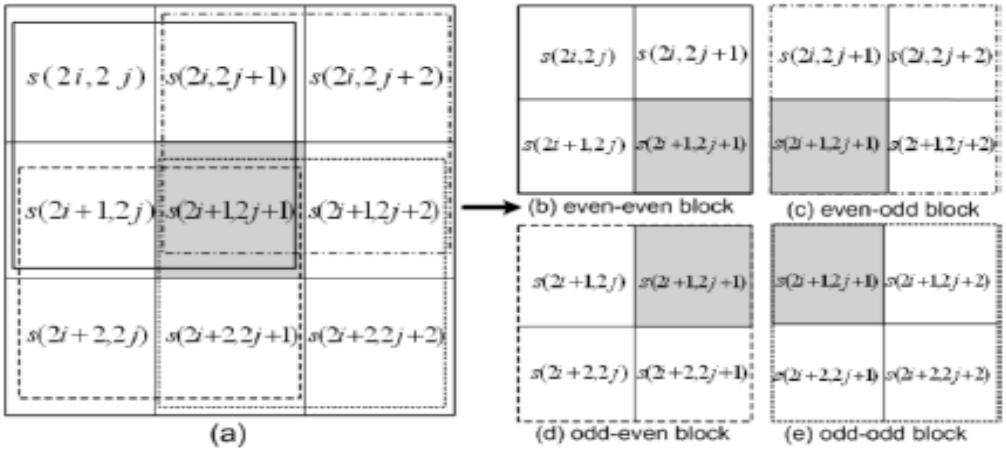


Figure 3. Four different 2 x 2 blocks in 3 x 3 blocks of binary image.

$$Bp(i,j) \in \begin{cases} \text{EEBs} & \text{for } (mod(x,2)=0) \wedge (mod(y,2)=0) \\ \text{EOBs} & \text{for } (mod(x,2)=0) \wedge (mod(y,2)=1) \\ \text{OEBs} & \text{for } (mod(x,2)=1) \wedge (mod(y,2)=0) \\ \text{OOBs} & \text{for } (mod(x,2)=0) \wedge (mod(y,2)=1) \end{cases} \qquad (20)$$

Where (x,y) denotes xth row and yth column of an image ,(i ,j) denotes the index of the current 2x2 block mod() denotes a modulo operation and " ^ " represents logical "AND"operation. Hence, three additional transforms, i.e., even-odd, odd-even and odd-odd, can be defined. These transforms, together with the even-even transform, are collectively called interlaced morphological binary wavelet transform (IMBWT).The operators for the even-odd transforms are , the signals obtained by applying the analysis operators for the even-odd transform are given by

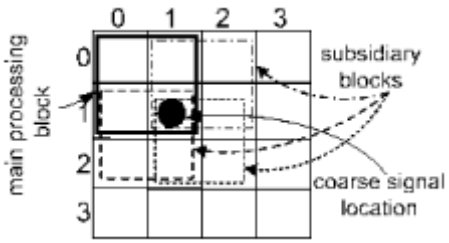


Figure. 4. Main processing block and its subsidiary block

$$S^{eo}(i,j) = \psi^{eo+}(S)\ (i,j) = S\ (2i+1,2j+2) \quad (21)$$

$$v^{eo}(i,j) = \varpi_v{}^{eo+}(S)\ (i,j) = S\ (2i+1,2j+1) \oplus S(2i+1,2j+2) \quad (22)$$

$$h^{eo}(i,j) = \varpi_h{}^{eo+}(S)\ (i,j) = S\ (2i,2j+2) \oplus S(2i+1,2j+2) \quad (23)$$

$$d^{eo}(i,j) = \varpi_d{}^{eo+}(S)\ (i,j) = S\ (2i,2j+1) \oplus S(2i+1,2j+1) \oplus S(2i,2j+2) \oplus S(2i+1,2j+2) \quad (24)$$

The odd-even and odd-odd transforms can be defined in the same way. For simplicity, we use, s_k, v_k, h_k, and d_k, $k \in \{ee,eo,oe,oo\}$ to represent the signals obtained from different transforms. There are four single processing cases (SPCs): even-even, even-odd, odd-even and odd-odd that are determinant on the main processing blocks to be EEBs, EOBs, OEBs and OOBs. Consider the even-even processing case where the main processing blocks are EEBs and EOBs, OEBs and OOBs which are interlaced with the EEBs are shown as the subsidiary blocks in Fig. 4. In Applying the IMBWT for data hiding in binary images, the processing of images is always based on 2 x 2 blocks (i.e., main processing blocks). However, the flippability of a coarse signal is determined by considering 3 x 3 blocks, which consist of both the main processing blocks and the subsidiary blocks.

III.I Single Processing

As flipping an edge pixel in binary images is equivalent to shifting the edge location horizontally one pixel and vertically one pixel. A horizontal edge exists if there is a transition between two neighboring pixels vertically and a vertical edge exists if there is a transition between two neighboring pixels horizontally. To define the flippability condition for a coarse signal, we actually consider the 3 x 3 block in such a way that the shifted edges can be tracked for the convenience of blind watermark extraction. The flippability condition (or cross condition) for a SPC is defined as follows: a coarse signal (in a main processing block) is flippable if both horizontal and vertical edges exist. Specifically, consider the even-even processing case, a horizontal edge exists if either $\boldsymbol{h_{ee}}$ or $\boldsymbol{h_{oe}}$ equals to 1 and a vertical edge exists if either or equals to 1. The *flippability* condition $f_{ee}(i,j)$ is given by

$$f_{ee}(i,j) = (v_{ee(i,j)} \oplus v_{eo(i,j)})) \wedge (h^{ee}(i,j) \oplus h^{eo}(i,j)) \quad (25)$$

$$f_{ee}(i,j) = 1. \quad (26)$$

Let us now consider a 3 x 3 block of an input image s. The number of transitions from the current candidate pixel (the center pixel) to its 4-neighbors in the horizontal and vertical directions is represented by $\mathbf{N_h}$ and $\mathbf{N_v}$, respectively, which are calculated from the center pixel to its 4-neighbors. Assume the center pixel S (2i+1, 2j+1) is [see Fig. 3(a)], and $\mathbf{N_h}$ and $\mathbf{N_v}$ are given by

$$N_h = \sum s(2i+1,2j+k) \oplus s(2i+1,2j+1+k) \quad (27)$$

$$N_v = \sum s(2i+k,2j+1) \oplus s(2i+1+k,2j+1) \quad (28)$$

By satisfying the cross condition that $f_{ee}(i,j)=1$ and N_v do not change when the center pixel is flipped. Hence, the 4-connectivity from the center pixel to its 4-neighbors is preserved. Further, the center pixel has a white 4-neighbor pixel in both horizontal and vertical directions. Similarly, the flippability conditions for the odd-odd, even-odd, and odd-even processing cases is calculated. In summary, the flippability conditions help identify all the candidate pixels for which the pixels directly above and below the candidate pixel have different colors ("0" and "1") and the pixels immediately to the left and right of the candidate pixel have different colors. In addition, in applying these flippability conditions for data hiding, we require that the two chosen candidate pixels should not be 4-adjacent, i.e., horizontally or vertically adjacent to each other to avoid poor visual quality. It can be seen that the flippability of the center pixel in each 3 x 3 block is independent of the center pixel value.

IV. High Capacity data Hiding Using Orthogonal Embedding

The process of orthogonal embedding consists of following processes.

IV.I Authentication Watermark Generation, Embedding and Verification

Firstly it is necessary to apply the flippability.

IV.I.I Flippability decision

In applying the IMBWT for data hiding in binary images, the processing of images is always based on 2X2 blocks (i.e., main processing blocks). The flippability conditions of a coarse signal, is determined by considering 3X3 blocks, which consist of both the main processing blocks and the subsidiary blocks

IV.I.II Embeddability

In order to generate the hard authenticator watermark, the key issue is how to locate the "embeddable" pixels given the watermarked image. The "embeddability" of a block depends on the "flippability" of the determined pixels in the block.

IV.I.III Watermark embedding

The watermark embedding process is summarized as follows.

1) Partition the image into equal size square blocks; note that the block size does not need to be square.
2) Determine the flippability of the determined pixels based on the "Flippability Criterion".
3) Once a pixel is identified as "flippable", the block is marked as "embeddable". The current "flippable" pixel is Identified as the "embeddable" pixel, i.e. "embeddable" location of the block.
4) Proceed to the next block.
5) Repeat steps 2 to 4 until all the blocks are processed.
6) Embed the watermark in the "embeddable" blocks by flipping the "embeddable" pixels.

IV.I.IV Authentication process

To address the security concerns, method proposes to generate the Authenticator watermark in fig. by encrypting the XORed value of the replicated hash value of the binary images and the payload watermark.

IV.I.V Verification process

Extract the watermark based on flippability & embeddable pixel criterion. Thus the tampering can be detected.

The proposal of processing an image in 2X 2 blocks not only prevents the computational load from getting high for blockbased approach with large block size but also improves the efficiency of utilization of flippable pixels., the flippability conditions help identify all the candidate pixels for which the pixels directly above and below the candidate pixel have different colors ("0" and "1") and the pixels immediately to the left and right of the candidate pixel have different colors. In addition, in applying these flippability conditions for data hiding, we require that the two chosen candidate pixels should not be 4-adjacent, i.e., horizontally or vertically adjacent to each other to avoid poor visual quality. It can be seen that the flippability of the center pixel in each 3X3 block is independent of the center pixel value.

IV.II Double Processing

The capacity of the proposed scheme is determined by the number of pixels that satisfy the flippability condition. This capacity can be increased significantly by combining the two single processing cases, namely, Double Processing Case (DPC). Based on the observations of (27) and (28), the flippability condition of the even-even processing case (i.e. $f_{ee}(i,j)$) is not affected by flipping the candidates in the odd-odd processing case (e.g., S (2i, 2j) and vice versa, as illustrated in Fig. 5. This "nonintersection" property of the two processing cases renders the processing of the even-even and odd-odd cases can be done together. The same applies to the even-odd and odd-even processing cases. For convenience, we call the two combined processing cases as a "Pair Case".

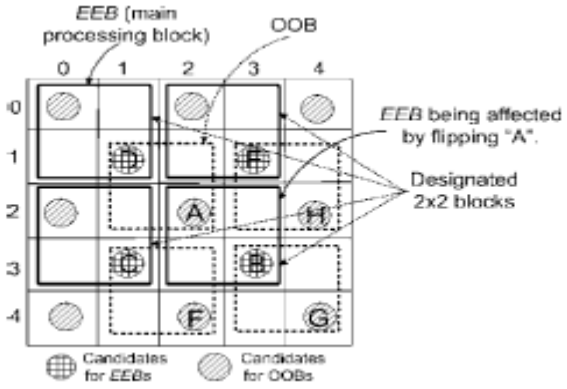


Figure.5. Designated 2 x 2 blocks.

As evidenced from the increase in the number of candidate pixels, i.e., from [1/4 x M x N] to [1/2 x M x N] for an image of size, the capacity can be approximately doubled by combining the two processing cases. This idea is motivated by the quantization index modulation based data-hiding method [18]. The even-even and odd-odd processing cases can be viewed as two orthogonal sets of embedders indexed at the 2 x 2 blocks starting from even-even and odd-odd coordinates, denoted as Q_{ee} and Q_{oo} , respectively.

In this paper, the embedders are an ensemble of the embedding functions of a "Pair Case", e.g., the embedding functions of even-even and odd-odd cases.

In this approach. We define a designated 2 x 2 block $D_b(i,j)$ as the 2 x 2 block that contains the two candidate pixels for a pair case, which is shown in Fig. 6. The designated 2 x2 block can be chosen from one of the two processing blocks of a pair case, but only one candidate pixel is chosen to hide data in each. $D_b(i,j)$ Hence, the maximum capacity $\mathbf{C_c}$ is upper bounded by $C_c <= [1/4 \times M \times N]$ To achieve higher security, the choice of an Embedder Q_k in each $\boldsymbol{D_b(i,j)}$ using DPC can depend on a random key $\mathcal{R}_s$ where $\mathcal{R}_s \in \{0,1\}$.. For example, for the m^{th} $D_b(i,j)$ block, we choose $\mathcal{Q}_{ee}$ when $\mathcal{R}_s(m)=1$ and choose $\mathcal{Q}_{oo}$ when $\mathcal{R}_s(m)=0$.. By choosing $\mathcal{Q}_{ee}$. we check $f_{ee}(i,j)$ first, mark the current candidate pixel as embeddable if $f_{ee}(i,j)=1$ and proceed to the next $D_b(i,j)$. Otherwise $f_{oo}(i,j)$,will be checked (i.e., when $f_{ee}(i,j) \neq 1$). Similarly, $f_{oo}(i,j)$ is checked first by choosing $\mathcal{Q}_{oo}$.. From the above discussion, it is noticed that there may be two candidate pixels being flipped in some 2 x 2 blocks (not $D_b(i,j)$), e.g., the OOBs in Fig.6.

V. Experimental Results

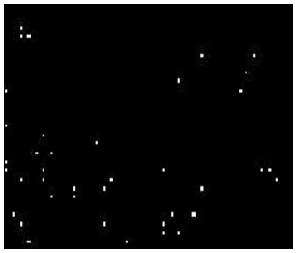
To show the capacity increase by employing DPC compared with that of SPC, 100 images of a variety of sizes were used. These images are of different types (e.g., cartoons, sketch, drawing, handwritten and text in different languages) and various dimensions (e.g., 132x132, 128 x 128,232x202, 720x756, and 450x524). The Capacities of the watermarked image, which is defined in Table 1. The increase in capacity represented in percentage from SPC to DPC.

The capacity mainly depends on the contents of the images, which varies for different types of images. The richer the contents of the images (more edges exist), the larger the capacity. For images of same size and similar contents, in general, text or cartoon images which consist of rich corners and thin strokes may easily cause larger distortion. This is due to the facts that few good patterns exist in corners and thin strokes, hence flipping pixels can be easily noticed. Following figures indicates the data hiding effects.

Authentication Results

Fig.6. (a) Original Image Of Size 132x132.Max. Data Hiding Capacity for SPC=104,DPC=256,

(b)Watermarked image with 80 bits embedded by employing SPC



(c)Difference Image with SPC flipped pixels

(d)Watermarked image with 40 bits embedded by employing DPC

(e)Difference Image with DPC flipped pixels

VI. Conclusion

This paper, present a high-capacity approach for data-hiding scheme for binary images based on the interlaced morphological binary wavelet transforms. The relationship between the coefficients obtained from different transforms is utilized to identify the suitable locations for watermark embedding such that blind watermark extraction can be achieved. Two processing cases that are not intersected with each other are employed for orthogonal embedding in such a way that not only can the capacity be significantly increased, but the visual distortion can also be minimized. Comparative results show that the present scheme is quite superior in being able to attain larger capacity while maintaining acceptable visual distortion and low computational cost.

References

[1] I. J. Cox, M. L. Miller, and J. A. Bloom, San Mateo, CA: Morgan Kaufmann,(2001), *Digital Watermarking.*

[2] B. Furht and D. Kirovski, B. Furht and D. Kirovski, Eds. Boca Raton, FL: CRC(2005), *Multimedia Security Handbook*.

[3] Y. Liu, J. Mant, E. Wong, and S. H. Low (1999), "Marking and detection of text documents using transform-domain techniques," in *Proc. SPIE*, San Jose, CA, vol. 3657, pp. 317–328.

[4] Q. Mei, E. K. Wong, and N. Memon (2001), "Data hiding in binary text document," in Proc. SPIE, vol. 4314, pp. 369–375.

[5] Y. C. Tseng and H.-K. Pan, "Data hiding in 2-color images (Jul. 2002)," *IEEE Trans. Comput.*, vol. 51, no. 7, pp. 873–878,.

[6] H. Lu, A. C. Kot, and Y. Q. Shi (Feb. 2004), "Distance-reciprocal distortion measure for binary document images," *IEEE Signal Process. Lett.* vol. 11,no. 2, pp. 228–231.

[7] H. Yang and A. C. Kot (Apr. 2007), "Pattern-based date hiding for binary images authentication by connectivity-preserving," *IEEE Trans. Multimedia*, vol. 9, no. 3, pp. 475–486.

[8] B. B. Zhu, M. D. Swanson, and A. H. Tewfik(Mar.2004) , "When seeing isn't believing," *IEEE Signal Process. Mag*, pp. 40–49.

[9] H. Lu, X. Shi, Y. Q. Shi, A. C. Kot, and L. Chen (2002), "Watermark embedding in DC components of DCT for binary images," in *Proc. IEEE Workshop on Multimedia Signal Processing*, Dec. 9–11, pp. 300–303.

[10] H. J. A. M. Heijmans and J. Goutsias (Nov. 2000), "Nonlinear multiresolution signal decomposition schemes-part I: Morphological wavelets," *IEEE Trans. Image Process.*, vol. 9, no. 11, pp. 1897–1913.

[11] M. Wu and B. Liu (Aug. 2004), "Data hiding in binary images for authentication and annotation," *IEEE Trans. Multimedia*, vol. 6, no. 4, pp. 528–538.
.

[12] H. Y. Kim and R. L. de Queiroz (2004) , "Alteration-locating authentication watermarking for binary images," in *Proc. Int. Workshop Digital Watermarking*, pp. 125–136.

[13] H. J. A. M. Heijmans and J. Goutsias (Nov. 2000), "Nonlinear multiresolution signal decomposition schemes-part II: Morphological wavelets," *IEEE Trans. Image Process.*, vol. 9, no. 11, pp. 1897–1913.

[14] "Digital Watermarking" available at

http://en.wikipedia.org/wiki/Digital_watermarking

[15] "Fundamentals of Wavelets" available at

http://documents.wolfram.com/applications/wavelet/index2.html

VII. Observation Table I

Comparison of data hiding Capacities for SPC and DPC

Image	SPC capacity	DPC Capacity	Increase in Capacity (In characters)
	Embed data in characters		
Lily.bmp	13	32	19
Text.bmp	0	15	15
Drawing.bmp	3	9	6
Tiger.bmp	159	351	192
Sparkle.bmp	2	9	7
Sketch.bmp	13	30	17
Letter.bmp	4	8	4
Graphics.bmp	6	15	9
Nano.bmp	90	191	101
Hill.bmp	56	150	94
Rangoli.bmp	17	43	26
Cartoon.bmp	189	429	240

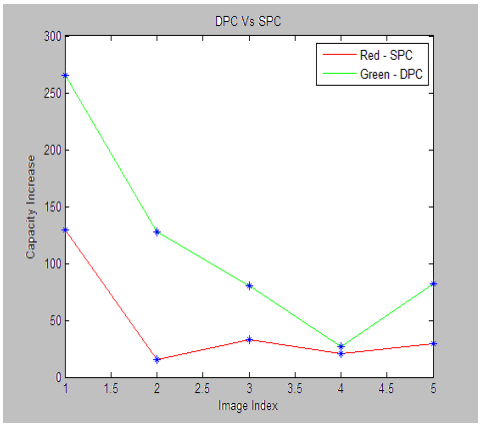


Figure.7. Graph- Capacity Increase Using SPC and DPC

The above fig.7 shows the graph of image index versus Data hiding capacities. It consists of 5 different class images. Observations shows that capacity has increased significantly using DPC versus SPC, e.g., increase from approximately 56% to 69% for the sample five test images.